

UNIVERSITY OF QUEENSLAND

Prentice Computer Centre

NEWSLETTER

authorization: Director of the Centre

1 CHANGE TO A NEW MONITOR

The Prentice Computer Centre is proceeding shortly to the implementation of a new operating system on the Centre's DECsystem-10. The new monitor will be based on the DEC version 6.02 (the second release of the 6-series monitors).

The motivation for the upgrade is four-fold:

- (1) DEC provide full support only for the latest versions of the software which they supply.
- (2) 6.02 is considered to have a number of improvements relevant to our installation.
 - (a) A more efficient scheduler and dispatcher.
 - (b) A much improved Magnetic-tape service routine.
 - (c) Improved handling of binary data on communications lines.
- (3) Phase II of the U.Q. communications network development is starting and is best based on the latest version of the monitor.
- (4) 6.02 is reputed to be the final or "stabilisation" version for KA-10 processors such as we have at U.Q.

The new 6.02 monitor is substantially the same as the existing monitor and most users should not be aware of the change. There are, however, some new features and changes that should be noted.

Users are requested to assess the changes and test any critical software or files under 6.02 before final installation (on or after Monday 13 September). To facilitate testing, a free computing service running 6.02 will be made available on two Saturdays before operational installation (4 and 11 September). Please report any difficulties to Doug Lennox or Chris de Voil.

Listed below is a more detailed description of the new facilities in 6.02:

1.1 Features of the 6.02 Monitor of importance to users

1.2 Device Names

Device names are formatted differently in the 6.02 monitor than in the existing monitor. All physical devices except disks and terminals have names of the form:

gggnnu

where: ggg is the generic device name (e.g. LPT, CDR)

nn is the 2-digit node number (e.g. 01, 10, 77)
Leading zeroes are required.

u is the 1-digit physical unit number in the range
0 through 7.

Device names can be specified as being logical names, physical names or generic names. Logical names take precedence over physical names. Examples of the lgeal physical device name formats are shown below:

Physical Device Name Formats

<u>Format</u>	<u>Example</u>	<u>Meaning</u>
dddn	LPT6	The monitor will attempt to select the device LPT6 at the user's current node/station.
dddnnu	LPT132	The monitor will attempt to select device LPT2 at node/station number 13.
ddd	LPT	The monitor will attempt to select a device of the desired type (e.g. a line printer) at the user's current node/station.
dddnn	LPT01	The monitor will attempt to select any line printer at node/station number 01.

Generic device names can be written as three-letter codes or two-letter codes. The acceptable generic device names are listed below. The existing 3-letter codes will therefore continue to work but users are urged to move to the two-letter codes as this is now the corporate Digital Equipment Corporation standard across their product line. Three-letter codes may not be supported in the future.

NOTE

Unit numbers may not be used
with 2-letter generic names.

System Devices

<u>Device</u>	<u>Generic Physical Device Name</u>	
	<u>3-letter</u>	<u>2-letter</u>
Card punch	CDP:	CP:
Card reader	CDR:	CR:
Console TTY	CTY:	not acceptable
DECtape	DTx:	DT:
Disk	DSK:	DS:
Packs	DPx:	DP:
Fixed-head	FHx:	FH:
Display	DIS:	not acceptable
Line printer	LPT:	LP:
lower-case		LL:
upper-case		LU:
Magnetic tape	MTx:	M7: (7-track) M9: (9-track) MT: (any unit)
Operator Terminal	OPR:	not acceptable
Paper tape punch	PTP:	PP:
Paper tape reader	PTR:	PR:
Plotter	PLT:	not acceptable
Pseudo-TTY	PTY:	not acceptable
System Library	SYS:	SY:
Terminal	TTY:	TT:

1.3 The Scheduler

The scheduler released with the 6.02 monitor is designed to co-ordinate system resource management to accomplish the following:

1. maximize system throughput
2. maximize system responsiveness
3. maximize CPU utilization
4. minimize system overhead

The scheduler allocates CPU time on the basis of a three-level system of queues. The queues are -

1. real-time queues for servicing jobs requiring real-time guaranteed access to the CPU
2. a fast response queue for servicing jobs coming out of TTY I/O wait or jobs just starting execution
3. a round robin queue (which is the main processing queue).

The round robin queue is divided into a number of ordered classes. Each class has a quota associated with it, which is defined by the system administrator. This quota is defined in terms of the percentage of the available system resources which will be allocated to jobs belonging to the associated class.

Initially, the system will be used with standard versions of the various scheduler controlling files. Experience at other installations has shown a need to experiment in this area with caution but it does offer a very powerful tool for the control of system utilization.

The scheduler collects information concerning all of its activities enabling the system administrator to monitor the effectiveness of the installation's external scheduling policy.

1.4 Maximum Number of Jobs

It is now possible to configure a system with up to 511 jobs! We will be staying below 100 for some time yet.

1.5 Enqueue/Dequeue

The Enqueue/Dequeue Facility can be used to share resources of a system with other users on a system. Examples of resources that may be controlled by this facility are:

1. files
2. operations on files (e.g. READ, WRITE)
3. records
4. devices
5. memory pages.

Its most common application is, however, the control of simultaneous updating of files by multiple users.

The Enqueue/Dequeue Facility insures data integrity among jobs when the jobs co-operate in their use of both the facility and the physical resource.

1.6 FILOP. Monitor Call

The FILOP. monitor call allows a user program to create, read, update, append to, or close a file, or the user program may optionally request the monitor to update a RIB of a file. The calling sequence for FILOP. is as follows:

```

        MOVE    ac, XWD[length,addr]
        FILOP.  ac,
                error return
                normal return

addr:   x,,function-code
        SIXBIT/device-name/
        output-buffer-header,,input-buffer header
        output-buffers,,input-buffers
        0,,LOOKUP/ENTER Block
        length,,PATH. Block

```

1.7 Packed Image Mode (PIM)

Packed image mode (octal code 2) is legal for TTY input and output except for pseudo-TTY's. PIM is designed for high efficiency character throughput between programs and external devices, which is accomplished by minimizing the monitor's character manipulation and testing.

In Packed Image Mode, characters are maintained as 8-bit quantities (i.e. 7 data bits and 1 parity bit), which are stored in buffers at the rate of 4 characters per word. The user program may set a "break" consisting of one to four break characters for each line INITed in Packed Image Mode. The break set is defined via the TRMOP. monitor call.

When the monitor receives a character from a Packed Image Mode line with a pending IN or INPUT, the character is compared to each field in the break set if no match occurs, the character is put in the buffer and the interrupt is dismissed. In the case where a match does occur, the character is put into the buffer, the buffer is then terminated and the controlling program is awakened. To avoid the possibility of a terminal getting stuck in PIM mode, and to allow for the case where the user wishes to be awakened on each character, the user program specifies an empty (0,,0) break set. In general operation, all characters, including control characters are passed by PIM with no monitor intervention (e.g. fill, CR/LF etc.) with the following exceptions: if PAGE mode is set, the characters XON and XOFF react in the normal PAGE mode sense.

1.8 The New Magnetic Tape Service Routine

TAPSER completely replaces MTXSER and TCASER. There should be no observable differences to existing user-mode mag-tape software. Non-DEC supplied mag-tape handling software is not guaranteed to be without compatibility problems.

The MTAPE UUO is fully supported and compatible with the existing monitor. All new features and functions will not be retrofitted into this UUO. Also all illegal functions will no longer "do something". Instead they will give an "illegal UUO..." message and terminate execution. (N.B. the MTAPE 200 function now applies to the TM10 sub-system only!).

✓ The MTCHR. and MTAID. UUO's are also fully supported and compatible with the existing monitor. The hardware error information fields (words 15-20) of the MTCHR. UUO are valid only on a TM10 sub-system.

The TAPOP. UUO is introduced in the 6.02 release. Among other things, it allows users to access the tape label data block.

In order to achieve compatibility with new tape sub-systems (e.g. TU70's), end-of-tape (IO.EOT) is no longer set in the I/O status bits returned by a GETSTS monitor call while doing input from a tape. In 5.07 (using a TM10) physical end of tape caused IO.EOT to be set even on a read. In particular, FRS V15 as distributed (unsupported) uses this 5.07 bug.

Since the magtape sub-systems now supported by the monitor have such vastly different hardware characteristics the "DEVSTS" monitor call will now always return a 0 for a magtape. If information about the tape status is desired, then the "MTCHR." or "TAPOP." monitor calls should be employed.

On a GET or RUN command from a magtape, if no extension is supplied or the extensions .SAV or .EXE are specified, the monitor assumes the file which it is inputting is in new (EXE) format on the magtape. To get an old format file (.SAV, .HGH etc.) from a magtape, a non-null extension which is neither .EXE or .SAV must be specified as well as a core argument. A core argument is not required if the file which has been SAVED on the magtape is in EXE format.

1.9 Detailed Documentation

Until such time as formal user documentation is available on new facilities, users wishing to use such facilities should obtain detailed information from the Program Librarian (Geoff Vandenberg). At this stage, we request your co-operation in limiting such requests to cases where you have specific requirement rather than general interest.

Doug Lennox and Chris de Voil will be pleased to hear from you on any difficulties.

2 COBOL - A CORRECTION

Despite the fact that Cobol V6A on NEW: had received extensive testing and in-house use since January; when we tried to make it the standard version on July 26, two problems were encountered in the use of the object-time system LIBOL. These problems were (a) special use of FUNCT. calls to obtain additional core chunks sometimes did not work and more seriously (b) incompatibilities existed between the calls generated for the new and old sort modules, so that a program compiled and saved under the old system would fail if it used the Sort module in the new operating system.

This latter problem would have required the recompilation of all previously saved programs and it was felt that this was not justified. Accordingly, we have reverted to use of the previous variant of Cobol V6A on STD: and will not attempt to implement the unbundled sort procedures within V6A.

The following is now the disposition of Cobol and Sort -

NEW: Cobol V10 with unbundled Sort
Sort V2

STD: Cobol V6A with CSORT
Sort V1A

OLD: No versions of Cobol or Sort.

This does mean that to take advantage of the unbundled sort, which testing has shown to give an advantage of between 1:2 or 1:5 for a representative range of files, users will have to use the version of Cobol on NEW:. It is currently expected that this version will become standard by 1 January 1977 and users are encouraged to recompile their programs with it. After its adoption, any existing SAVEd programs will continue to use the older version of the operating system which will be maintained for a limited period. All new work will, however, use the new version of the operating system.

3 CARD AND PAPER TAPE PUNCHING

Contrary to what might have been implied in newsletter N-202, there is no reduction in charging rate for lower than standard priorities as regards card or paper tape punching. These particular activities require substantial manual involvement and the amount of punching of either media usually justifies at most a weekly run of the system program concerned. There is, therefore, no case for a "low priority" rate and little justification in committing effort for enhancements of this system.

4 ESSEX BCPL

A new Essex BCPL system is now available, but at this stage is not supported by the Centre. To invoke the compiler type:

```
.R BCL:BCPL
*PROG                      ; /K for upper/lower case
.
.
*+C
.EX/REL PROG
```

The .BCPL command still invokes the U.Q. supported version.

```
.HELP BCPL                ; about the U.Q. version
.HELP EBCPL                ; about the Essex version
```

Mr. A. Lister of the Department of Computer Science has available for perusal a copy of the Essex manual.

5 'CLEAN-UP' OF THE OFFLINE AREA

In order to reduce the offline area to reasonable proportions, we will delete all files with the extension of BAK, TMP, MAP, LOG, LPT and PLT. We hope that now the FMS system has been transferred from RPO2s to RPO3s and the above files have been deleted, it will relieve the necessity of skimming the offline areas on a six monthly basis.

* * * * *